

1. Create a simple form in HTML that includes a text input, a password input, and a submit button.
2. How do you create a hyperlink in HTML that opens in a new tab?
3. Write HTML to create a table with 3 rows and 3 columns. The first row should contain headers.
4. Create an ordered list and an unordered list in HTML.
5. How do you apply a background color to the entire page in CSS?
6. Write CSS to style an element with the class "highlight" to have yellow background and bold text.
7. Write CSS to change the color of all `<h1>` elements to blue.
8. Write CSS to add a 10px padding, 5px border, and 15px margin to a `<div>` element.
9. Create a flex container with three child elements that are evenly spaced.
10. Explain the purpose of HTML5 semantic elements and provide examples of at least five semantic elements.
11. Demonstrate how to use the HTML5 Geolocation API to get the user's current location.
12. Create a CSS Grid layout with three columns and three rows where the first row spans all columns and the last column spans all rows.
13. Write CSS to create a button that changes color smoothly on hover using transitions.
14. Create a CSS animation that moves a `<div>` from left to right.
15. Explain how to use CSS variables and provide an example.
16. Write a media query that changes the background color of the body to lightgray when the viewport width is 600px or less.
17. Write a media query to apply a different font size for a `<p>` element when the device is in landscape orientation.
18. Write a media query to target devices with high-resolution (Retina) displays.
19. Write a media query that applies styles only when the viewport width is between 768px and 1024px and the orientation is portrait.
20. Write a set of media queries to create a responsive grid layout with 4 columns on large screens, 2 columns on medium screens, and 1 column on small screens.
21. Write a media query to apply styles specifically for printed documents.
22. Write a media query to adjust the padding of a header element based on the viewport height.
23. Write a media query to change the layout of a div to a flexbox when the aspect ratio is 16:9.
24. Write a media query that applies a dark theme when the user's system preference is set to dark mode.
25. Write a media query to apply styles only on devices that support hover functionality.
26. Write CSS to position an element 10px from the top and 20px from the left of its normal position.
27. Write CSS to position an element at the bottom-right corner of its nearest positioned ancestor.
28. Explain what the default value of the `position` property is and how it affects the placement of an element.

29. Write CSS to create a fixed navigation bar that stays at the top of the viewport when scrolling.
30. Explain the `position: sticky` property and provide an example where a header becomes sticky when scrolling.
31. Write CSS to position three overlapping elements using `z-index` such that the middle element appears on top.
32. Write CSS to absolutely position an element inside a scrolling container.
33. Write CSS to center an absolutely positioned element within its positioned ancestor.
34. Write CSS to create a fixed footer that stays at the bottom of the viewport.
35. Write CSS to vertically and horizontally center a flex item within a flex container.
36. Write CSS to allow flex items to wrap onto multiple lines if there is not enough space in a single line.
37. Write CSS to align flex items to the start of the cross-axis (top in a row layout).
38. Write CSS to change the order of flex items such that the third item appears first.
39. Write CSS to allow flex items to grow to fill the remaining space but shrink if necessary to fit within the container.
40. Write CSS to align a specific flex item to the end of the cross-axis, regardless of the alignment of other items.
41. Write CSS to create a nested flex container where the outer container arranges items in a row and the inner container arranges items in a column
42. Write CSS to create a responsive flex container that switches from a row layout to a column layout on screens smaller than 600px.